On State-Dominance Criteria in Fork-Decoupled Search

Álvaro Torralba and **Daniel Gnad** and **Patrick Dubbert** and **Jörg Hoffmann**
Saarland University
Saarbrücken, Germany
{torralba, gnad, hoffmann}@cs.uni-saarland.de; s9padubb@stud.uni-saarland.de

Abstract

Fork-decoupled search is a recent approach to classical planning that exploits *fork* structures, where a single *center* component provides preconditions for several *leaf* components. The *decoupled states* in this search consist of a center state, along with a *price* for every leaf state. Given this, when does one decoupled state dominate another? Such *state-dominance* criteria can be used to prune dominated search states. Prior work has devised only a trivial criterion. We devise several more powerful criteria, show that they preserve optimality, and establish their interrelations. We show that they can yield exponential reductions. Experiments on IPC benchmarks attest to the possible practical benefits.

1 Introduction

Fork-decoupled search is a new approach to state-space decomposition in classical planning, recently introduced by Gnad and Hoffmann [2015]. The approach partitions the state variables into disjoint subsets, *factors*, like in factored planning (e. g. [Amir and Engelhardt, 2003; Kelareva *et al.*, 2007; Fabre *et al.*, 2010; Brafman and Domshlak, 2013]). While factored planning is traditionally designed to handle arbitrary cross-factor interactions, fork-decoupling assumes these interactions to take a fork structure [Katz and Domshlak, 2008; Katz and Keyder, 2012; Aghighi *et al.*, 2015], where a single *center* provides preconditions for several *leaves*. A simple pre-process can determine whether such a fork structure exists, and extract a corresponding factoring if so.

Fork factorings identify a form of "conditional independence" between the leaf factors: Given a fixed center path π^C, the *compliant* leaf moves – those leaf moves enabled by the preconditions supplied along π^C – can be selected independently for each leaf. The decoupled search thus searches only over center paths π^C. Each *decoupled state* in the search represents the compliant leaf moves in terms of a *pricing function*, mapping each leaf-factor state s^L to the cost of a cheapest π^C-compliant path achieving s^L. As Gnad and Hoffmann (henceforth: GH) show, this can exponentially reduce state space size. It may also cause exponential blow-ups though.

The worst-case exponential blow-ups result from irrelevant distinctions in pricing functions. One means to combat this, and more generally to improve search, is *dominance pruning*, pruning a state $s^\mathcal{F}$ if a better state $t^\mathcal{F}$ has already been seen. But, given the complex structure of decoupled states, when is one "better" than another? GH employ the trivial criterion, where $s^\mathcal{F}$ and $t^\mathcal{F}$ must have the same center state and $t^\mathcal{F}$ needs to have cheaper prices than $s^\mathcal{F}$ for all leaf states. Here we introduce advanced methods, analyzing the structure of decoupled states to identify (and then, disregard) irrelevant distinctions. We devise several such methods, using different sources of information. We show that the methods preserve optimality, and we characterize their relative pruning power. We show that they can yield exponential search reductions. Experiments on International Planning Competition (IPC) benchmarks attest to the possible practical benefits.

The main text only outlines our proof arguments. The full proofs are in Appendix A.

2 Background

We use finite-domain state variables [Bäckström and Nebel, 1995; Helmert, 2006]. A *planning task* is a tuple $\Pi = \langle \mathcal{V}, \mathcal{A}, I, G \rangle$. $\mathcal{V}$ is a set of *variables*, each associated with a finite domain $\mathcal{D}(v)$. I is the *initial state*. The *goal* G is a partial assignment to $\mathcal{V}$. $\mathcal{A}$ is a finite set of *actions*, each a triple $\langle \mathsf{pre}(a), \mathsf{eff}(a), \mathsf{cost}(a) \rangle$ of *precondition*, *effect*, and *cost*, where $\mathsf{pre}(a)$ and $\mathsf{eff}(a)$ are partial assignments to $\mathcal{V}$, and $\mathsf{cost}(a) \in \mathbb{R}^{0+}$. For a partial assignment p, we denote with $\mathcal{V}(p) \subseteq \mathcal{V}$ the subset of variables on which p is defined. For $V \subseteq \mathcal{V}(p)$, we denote with $p[V]$ the assignment to V made by p. We identify (partial) variable assignments as sets of variable/value pairs, written as (var, val). A *state* is a complete assignment to $\mathcal{V}$. Action a is *applicable* in state s if $\mathsf{pre}(a) \subseteq s$. Applying a in s changes the value of all $v \in \mathcal{V}(\mathsf{eff}(a))$ to $\mathsf{eff}(a)[v]$, and leaves s unchanged elsewhere. We will sometimes write $s \xrightarrow{a} t$ for a transition from s to t with action a. A *plan* for Π is an action sequence π iteratively applicable in I which results in a state s_G where $G \subseteq s_G$. The plan is *optimal* if its summed-up cost, denoted $\mathsf{cost}(\pi)$, is minimal among all plans for Π.

We next give a recap of GH's definitions. A *fork factoring* $\mathcal{F}$ is a partition of $\mathcal{V}$ identifying a fork structure. Namely, (i) every action $a \in \mathcal{A}$ affects (touches in its effect) exactly one element (*factor*) of $\mathcal{F}$, which we denote $F(a)$. And (ii) there is a *center* $F^C \in \mathcal{F}$ s.t., for every $a \in \mathcal{A}$, $\mathcal{V}(\mathsf{pre}(a)) \subseteq$

$F^C \cup F(a)$. We refer to the factors $F^L \in \mathcal{F}^L := \mathcal{F} \setminus \{F^C\}$ as *leaves*. We refer to actions affecting F^C as *center actions*, and to actions affecting a leaf as *leaf actions*. By construction (each action affects only one factor) these two kinds of actions are disjoint. Center actions are preconditioned only on F^C, leaf actions may be preconditioned on F^C and the leaf they affect. In brief: *the center provides preconditions for the leaves, and there are no other cross-factor interactions.*

As a running example, we use a Logistics-style planning task with a truck variable t, a package variable p, and n locations $l_1, \dots, l_n$. $I = \{(t, l_1), (p, l_1)\}$ and $G = \{(p, l_2)\}$. Action *drive*(x, y) moves the truck from any location x to any other location y. The package can be loaded/unloaded at any location x with actions *load*(x)/*unload*(x) respectively. Then $\mathcal{F} = \{\{t\}, \{p\}\}$ is a fork factoring where $\{t\}$ is the center and $\{p\}$ is the single leaf. If we have m packages p_i, we can set each $\{p_i\}$ as a leaf.

Not every task Π has a fork factoring. GH analyze Π's causal graph (e. g. [Knoblock, 1994; Jonsson and Bäckström, 1995; Brafman and Domshlak, 2003; Helmert, 2006]) in a pre-process, identifying a fork factoring if one exists, else *abstaining* from solving Π. We follow this approach here. In what follows, we assume a fork factoring $\mathcal{F}$. Variable assignments to F^C are called *center states*, and for each $F^L \in \mathcal{F}^L$ assignments to F^L are *leaf states*. We denote by S^L the set of all leaf states, across $F^L \in \mathcal{F}^L$. For each leaf, s^L_I denotes the initial leaf state. For simplicity (wlog), we will assume that every leaf has a single goal leaf state, s^L_G.

Decoupled search searches over sequences of center actions π^C, called *center paths*, that are applicable to I. For each π^C, it maintains a compact representation of the *leaf paths* π^L that *comply* with π^C. A leaf path is a sequence of leaf actions applicable to I when ignoring preconditions on F^C. Intuitively, given the fork structure, a fixed center path determines what each leaf can do (independently of all other leaves, as they interact only via the center). This is captured by the notion of compliance: π^L complies with π^C if it uses only the center preconditions supplied along π^C, i. e., if π^L can be scheduled alongside π^C s.t. the combined action sequence is applicable in I. Decoupled search goes forward from I until it finds a center path π^C to a center goal state where every leaf has a π^C-compliant leaf path π^L to its goal leaf state. The global plan then results from augmenting π^C with the paths π^L.

In detail: A *decoupled state* $s^\mathcal{F}$ is given by a center path $cp(s^\mathcal{F})$. Its center state $cs(s^\mathcal{F})$ and *pricing function* $prices(s^\mathcal{F}) : S^L \mapsto \mathbb{R}^{0+}$ are induced by $cp(s^\mathcal{F})$, as follows. $cs(s^\mathcal{F})$ is the outcome of applying $cp(s^\mathcal{F})$ to s^L_I. $prices(s^\mathcal{F})$ maps each leaf state s^L to the cost of a cheapest $cp(s^\mathcal{F})$-compliant leaf path ending in s^L (or ∞ if no such path exists).[1] The *initial decoupled state* $I^\mathcal{F}$ has the empty center path $cp(I^\mathcal{F}) = \langle\rangle$. A *goal decoupled state* $s^\mathcal{F}_G$ is one with a *goal center state* $cs(s^\mathcal{F}_G) \supseteq G[F^C]$ and where, for every leaf factor $F^L \in \mathcal{F}^L$, its goal leaf state s^L_G has been reached, i. e., $prices(s^\mathcal{F}_G)[s^L_G] < \infty$. The actions applicable in $s^\mathcal{F}$ are those center actions a where $\mathsf{pre}(a) \subseteq cs(s^\mathcal{F})$. Applying a to $s^\mathcal{F}$ results in $t^\mathcal{F}$ where $cp(t^\mathcal{F}) := cp(s^\mathcal{F}) \circ \langle a \rangle$, inducing $cs(t^\mathcal{F})$ and $prices(t^\mathcal{F})$ as above.

In the running example, $cs(I^\mathcal{F}) = \{(t, l_1)\}$, $prices(I^\mathcal{F})[(p, l_1)] = 0$, $prices(I^\mathcal{F})[(p, t)] = 1$, and $prices(I^\mathcal{F})[(p, l_i)] = \infty$, for all $i \neq 1$. Observe that $prices(I^\mathcal{F})[(p, t)]$ represents the cost of a *possible* package move, not a move we have already committed to. The actions applicable to $I^\mathcal{F}$ are *drive*(l_1, l_i). Applying any such action, in the outcome decoupled state $s^\mathcal{F}$ we have $prices(s^\mathcal{F})[(p, l_i)] = 2$, while all other prices remain the same. If we apply *drive*(l_1, l_2), then $s^\mathcal{F}$ is a goal decoupled state. The global plan is then extracted from $s^\mathcal{F}$ by augmenting the center path $cp(s^\mathcal{F}) = \langle$*drive*$(l_1, l_2)\rangle$ with the compliant goal leaf path $\langle$*load*(l_1),*unload*$(l_2)\rangle$.

A *completion plan* for $s^\mathcal{F}$ consists of a center path π^C leading from $s^\mathcal{F}$ to some goal center state, augmented with goal leaf paths compliant with $cp(s^\mathcal{F}) \circ \pi^C$. That is, we collect the postfix path for the center, and the complete path for each leaf. The *completion cost* of $s^\mathcal{F}$, denoted $h^{\mathcal{F}*}(s^\mathcal{F})$, is defined as the cost of a cheapest completion plan for $s^\mathcal{F}$. By $d^{\mathcal{F}*}(s^\mathcal{F})$, we denote the minimum, over all optimal completion plans π^F, of the number of center actions (decoupled-state transitions) in π^F.

3 Decoupled State Dominance

A binary relation $\preceq$ over decoupled states is a *decoupled dominance relation* if $s^\mathcal{F} \preceq t^\mathcal{F}$ implies that $h^{\mathcal{F}*}(s^\mathcal{F}) \geq h^{\mathcal{F}*}(t^\mathcal{F})$ and $d^{\mathcal{F}*}(s^\mathcal{F}) \geq d^{\mathcal{F}*}(t^\mathcal{F})$. In *dominance pruning*, given such a relation $\preceq$, we prune a state $s^\mathcal{F}$ at generation time if we have already seen another state $t^\mathcal{F}$ (i. e., $t^\mathcal{F}$ is in the open or closed list) such that $s^\mathcal{F} \preceq t^\mathcal{F}$ and $g(s^\mathcal{F}) \geq g(t^\mathcal{F})$. Intuitively, $t^\mathcal{F}$ dominates $s^\mathcal{F}$ if it has an at least equally good completion plan and center path. The center path condition is needed only in the presence of 0-cost actions, and ensures that the completion plan for $t^\mathcal{F}$ does not have to traverse $s^\mathcal{F}$. If $t^\mathcal{F}$ can be reached with equal or better g-cost, pruning $s^\mathcal{F}$ preserves completeness and optimality of the search algorithm.

We derive practical decoupled dominance relations by efficiently testable sufficient criteria. The relations differ in terms of their pruning power. We capture their relative power with two simple terms of two simple notions. First, we say that $\preceq'$ *subsumes* $\preceq$ if $\preceq' \supseteq \preceq$, i. e., if $\preceq'$ recognizes every occurrence of dominance recognized by $\preceq$. Second, we say that $\preceq'$ is *exponentially separated* from $\preceq$ if there exists a family of planning tasks in which the decoupled state space is exponential in the size of the input task under dominance pruning using $\preceq$ and polynomial when using $\preceq'$.[2] We will devise several decoupled dominance relations, weaker and stronger ones. Weaker relations are useful in practice (only) when they cause less computational overhead.

Previous work only considered what we will refer to as the *basic* decoupled dominance relation, denoted $\preceq_B$.

Definition 1 ($\preceq_B$ relation) *$\preceq_B$ is the relation over decoupled states defined by $s^\mathcal{F} \preceq_B t^\mathcal{F}$ iff $cs(s^\mathcal{F}) = cs(t^\mathcal{F})$ and, for all $s^L \in S^L$, $prices(s^\mathcal{F})[s^L] \geq prices(t^\mathcal{F})[s^L]$.*

[1] Pricing functions can be maintained in time low-order polynomial in the size of the individual leaf state spaces. See GH for details.

[2] More precisely, as the pruning depends on the expansion order: in which this statement is true for any expansion order.

This method simply does a point-wise comparison between $prices(s^{\mathcal{F}})$ and $prices(t^{\mathcal{F}})$, whenever both have the same center state. Basic dominance pruning often helps to reduce search effort, but is unnecessarily restrictive in its insistence on *all* leaf prices being cheaper. This is inappropriate in cases where $s^{\mathcal{F}}$ has some irrelevant cheaper prices. It may, indeed, cause exponential blow-ups as, e. g., in our running example.

The standard state space in our running example is small, since $|\mathcal{V}| = 2$. Yet the decoupled state space has size exponential in the number n of locations. Through the leaf state prices, the decoupled states "remember" the locations visited by the truck in the past. For example, the decoupled state reached through the center sequence $\langle drive(l_1, l_3), drive(l_3, l_4)\rangle$ has finite prices for (p, l_1), (p, t), (p, l_3), and (p, l_4), and price ∞ elsewhere; while the decoupled state reached through the sequence $\langle drive(l_1, l_4)\rangle$ has finite prices for (p, l_1), (p, t), and (p, l_4). Intuitively, the difference between the two pricing functions does not matter, because, with initial location l_1 and goal location l_2, the prices for $(p, l_i), i > 2$ are irrelevant. But without recognizing this fact, the decoupled state space enumerates (pricing functions corresponding to) every combination of visited locations.

It is remarkable here that the blow-up occurs in a simple Logistics task. This is a new insight. GH already pointed out the risk of blow-ups, but only in complex artificial examples. On IPC benchmarks, empirically the decoupled state space always is smaller than the standard one. Our insight here is that this is not because blow-ups don't occur, but because the blow-ups (e. g. remembering truck histories) are hidden behind the gains (e. g. not enumerating combinations of package locations). Indeed, in the standard IPC Logistics benchmarks, the blow-up above occurs for all non-airport locations within every city, and these blow-ups multiply across cities. All our advanced dominance pruning methods get rid of this blow-up (though none guarantees to avoid blow-ups in general).

4 Frontier-Based Dominance

Our first dominance relation is based on the idea that differing prices on a leaf state s^L do not matter if "s^L has no purpose". In our running example, say that we are checking whether $s^{\mathcal{F}} \preceq t^{\mathcal{F}}$ and $prices(s^{\mathcal{F}})[(p, l_3)] = 2$ while $prices(t^{\mathcal{F}})[(p, l_3)] = \infty$, and thus $s^{\mathcal{F}} \not\preceq_B t^{\mathcal{F}}$. However, say that $prices(s^{\mathcal{F}})[(p, t)] = 1$. Then the cheaper price for (p, l_3) in $s^{\mathcal{F}}$ does not matter, because the only purpose of having the package at l_3 is to load it into the truck. Indeed, the only outgoing transition of the leaf state (p, l_3) leads to (p, t).

We capture the relevant leaf states in $s^{\mathcal{F}}$ in terms of its *frontier*: those leaf states that are either themselves relevant (this applies only to the goal leaf state), or that can still contribute to achieving cheaper prices somewhere.

Definition 2 (Frontier) *We define the* frontier *of a decoupled state $s^{\mathcal{F}}$, $F(s^{\mathcal{F}}) \subseteq S^L$ as $F(s^{\mathcal{F}}) := \{s^L_G\} \cup \{s^L \mid \exists s^L \xrightarrow{a} t^L : prices(s^{\mathcal{F}})[s^L] + \mathsf{cost}(a) < prices(s^{\mathcal{F}})[t^L]\}$.*

We now obtain a decoupled dominance relation by comparing prices only on the frontier of $s^{\mathcal{F}}$:

Definition 3 ($\preceq_F$ relation) *$\preceq_F$ is the relation over decoupled states defined by $s^{\mathcal{F}} \preceq_F t^{\mathcal{F}}$ iff $cs(s^{\mathcal{F}}) = cs(t^{\mathcal{F}})$ and, for all $s^L \in F(s^{\mathcal{F}})$, $prices(s^{\mathcal{F}})[s^L] \geq prices(t^{\mathcal{F}})[s^L]$.*

Theorem 1 *$\preceq_F$ is a decoupled dominance relation.*

Comparing the prices on the frontier is enough because, in any completion plan for $s^{\mathcal{F}}$, if a compliant leaf path π^L decreases the price of the goal leaf state (e. g., from ∞ to some finite value), then π^L must pass through a frontier state s^L. Hence, in a completion plan for $t^{\mathcal{F}}$, we can use the postfix behind s^L. This completion plan can only be better than that for $s^{\mathcal{F}}$ because $prices(s^{\mathcal{F}})[s^L] \geq prices(t^{\mathcal{F}})[s^L]$.

It is easy to see that $\preceq_F$ is strictly better than $\preceq_B$:

Theorem 2 *$\preceq_F$ subsumes $\preceq_B$ and is exponentially separated from it.*

The first part of this claim is trivial as both relations are based on comparing prices, but $\preceq_F$ does so on a subset of leaf states. A task family demonstrating the second part of the claim is our running example. The only leaf action applicable in any leaf state (p, l_i) is $load(l_i)$, leading to (p, t). However, for any reachable $s^{\mathcal{F}}$, we have $prices(s^{\mathcal{F}})[(p, t)] = 1$ because this price is already achieved in the initial state, and prices can only decrease. So the only possible frontier state, apart from (p, t), is the goal (p, l_2). But only two different prices are reachable for (p, l_2), namely ∞ and 2. This shows the claim.

5 Effective-Price Dominance

Our next method appears orthogonal to frontier-based dominance at first sight, but turns out to subsume it. The method is based on replacing the prices in $t^{\mathcal{F}}$, i. e., the dominating state in the comparison $s^{\mathcal{F}} \preceq t^{\mathcal{F}}$, with smaller *effective* prices, denoted $Eprices(t^{\mathcal{F}})$. We then simply compare all such prices:

Definition 4 ($\preceq_E$ relation) *$\preceq_E$ is the relation over decoupled states defined by $s^{\mathcal{F}} \preceq_E t^{\mathcal{F}}$ iff $cs(s^{\mathcal{F}}) = cs(t^{\mathcal{F}})$ and, for all $s^L \in S^L$, $prices(s^{\mathcal{F}})[s^L] \geq Eprices(t^{\mathcal{F}})[s^L]$.*

The modified comparison is sound because the effective prices are designed to preserve $h^{\mathcal{F}*}(t^{\mathcal{F}})$. Precisely: *(*) For any center path π^C starting in $t^{\mathcal{F}}$, and for any leaf state s^L of leaf F^L, if π^L_s is a π^C-compliant leaf path from s^L to s^L_G, then there exists a path π^L from s^L_I to s^L_G that complies with $cp(t^{\mathcal{F}}) \circ \pi^C$ such that* $\mathsf{cost}(\pi^L) \leq Eprices(t^{\mathcal{F}})[s^L] + \mathsf{cost}(\pi^L_s)$. In other words, if $prices(t^{\mathcal{F}})[s^L] > Eprices(t^{\mathcal{F}})[s^L]$, then any completion plan can be modified to use some other leaf state which does provide a total price of $Eprices(t^{\mathcal{F}})[s^L] + \mathsf{cost}(\pi^L_s)$ or less.

It turns out that this can be ensured with the following simple definition. We define $Eprices(t^{\mathcal{F}})$ as the point-wise minimum pricing function p that satisfies:

$$p[s^L] = \begin{cases} prices(t^{\mathcal{F}})[s^L] & \text{if } s^L = s^L_G \\ \min\{prices(t^{\mathcal{F}})[s^L], & \\ \quad \max_{s^L \xrightarrow{a} t^L} \left(p[t^L] - \mathsf{cost}(a)\right)\} & \text{otherwise} \end{cases}$$

For each F^L, $Eprices(t^{\mathcal{F}})$ can be computed by a simple backwards algorithm starting at the goal leaf state s^L_G. To illustrate the definition, consider any $t^{\mathcal{F}}$ in our running example. The price of (p, t) is 1, and its effective price also is 1 because its successor leaf state $s^L_G = (p, l_2)$ always has effective price ≥ 2. For any irrelevant location l_i, $i > 2$, however, due to the transition to (p, t) whose effective price is 1, we get $Eprices(t^{\mathcal{F}})[(p, l_i)] = 0$ regardless of what the actual price of (p, l_i) in $t^{\mathcal{F}}$ is. The effective price 0 is sound because, in any completion plan for $t^{\mathcal{F}}$ starting with $load(l_i)$, we can use $load(l_1)$ instead to get (p,t) with price 1.

Theorem 3 *$\preceq_E$ is a decoupled dominance relation.*

To prove Theorem 3, observe that, whenever $s^{\mathcal{F}} \preceq_E t^{\mathcal{F}}$, given a completion plan for $s^{\mathcal{F}}$, we can construct an equally good completion plan for $t^{\mathcal{F}}$ by using the same center path π^C, and, with (*) above, constructing equally good or cheaper compliant goal leaf paths. It remains to prove (*). Consider any $t^{\mathcal{F}}$, center path π^C, leaf state s^L, and π^C-compliant goal leaf path π^L_s starting in s^L. In our example, e. g., say $t^{\mathcal{F}}$ is reached from $I^{\mathcal{F}}$ by applying $drive(l_1, l_3)$; that $\pi^C = \langle drive(l_3, l_2)\rangle$; that $s^L = (p, l_3)$; and that $\pi^L_s = \langle load(l_3), unload(l_2)\rangle$. Then, exists $\pi^L = \langle load(l_1), unload(l_2)\rangle$ that is compliant with $cp(t^{\mathcal{F}}) \circ \pi^C$.

Formally, denote $\pi^L_s = \langle a_1, \dots, a_n\rangle$ and denote the leaf states it traverses by $s^L = s^L_0, \dots, s^L_n = s^L_G$. Observe that, as $Eprices(t^{\mathcal{F}})[s^L_n] = prices(t^{\mathcal{F}})[s^L_n]$, π^L_s necessarily passes through a leaf state s^L_i whose effective and actual prices in $t^{\mathcal{F}}$ are identical. Let i be the smallest index for which that is so. Then, for all $j < i$, $Eprices(t^{\mathcal{F}})[s^L_j] \neq prices(t^{\mathcal{F}})[s^L_j]$, and thus by the definition of effective prices we have that $Eprices(t^{\mathcal{F}})[s^L_j] \geq Eprices(t^{\mathcal{F}})[s^L_{j+1}] - \mathsf{cost}(a_{j+1})$. Accumulating these inequalities, we get *(**)* $Eprices(t^{\mathcal{F}})[s^L_0] \geq Eprices(t^{\mathcal{F}})[s^L_i] - \sum_{j=1}^{i} \mathsf{cost}(a_j)$. Consider now the path π^L from s^L_I to s^L_G constructed as the concatenation of: a cheapest $cp(t^{\mathcal{F}})$-compliant path to s^L_i (in our example, $\langle load(l_1)\rangle$); with the postfix of π^L_s behind s^L_i (in our example, $\langle unload(l_2)\rangle$). Then $\mathsf{cost}(\pi^L) = prices(t^{\mathcal{F}})[s^L_i] + \sum_{j=i+1}^{n} \mathsf{cost}(a_j)$. As $Eprices(t^{\mathcal{F}})[s^L_i] = prices(t^{\mathcal{F}})[s^L_i]$, we get $\mathsf{cost}(\pi^L) = Eprices(t^{\mathcal{F}})[s^L_i] + \sum_{j=i+1}^{n} \mathsf{cost}(a_j)$. With (**), we get the desired property that $\mathsf{cost}(\pi^L) \leq Eprices(t^{\mathcal{F}})[s^L_0] + \sum_{j=1}^{i} \mathsf{cost}(a_j) + \sum_{j=i+1}^{n} \mathsf{cost}(a_j) = Eprices(t^{\mathcal{F}})[s^L] + \mathsf{cost}(\pi^L_s)$, concluding the proof.

Theorem 4 *$\preceq_E$ subsumes $\preceq_F$ and is exponentially separated from it.*

To prove the exponential separation, we extend our running example with a *teleport*(l_i, l_j) action, for $i, j > 2$, that moves the package between irrelevant locations if the truck is at l_2. Then, as long as l_2 and at least one such l_i have not been visited yet, all leaf states (p, l_i) for $i > 2$ with finite price are in the frontier, and $\preceq_F$ suffers from the same blow-up as $\preceq_B$. The effective prices of (p, l_i), however, remain 0 as before.

To see that $\preceq_E$ subsumes $\preceq_F$, observe that the former can be viewed as a recursive version of the latter, when reformulating the frontier condition to "$\exists s^L \xrightarrow{a} t^L : p[s^L] < p[t^L] - \mathsf{cost}(a)$". Formally, one can show that, if $Eprices(t^{\mathcal{F}})[s^L] \leq prices(s^{\mathcal{F}})[s^L]$ holds for all frontier states $s^L \in F(s^{\mathcal{F}})$, then it also holds for all non-frontier states $s^L \notin F(s^{\mathcal{F}})$. This shows the claim as, for $s^{\mathcal{F}} \preceq_F t^{\mathcal{F}}$, we have $prices(s^{\mathcal{F}})[s^L] \geq prices(t^{\mathcal{F}})[s^L]$ on $s^L \in F(s^{\mathcal{F}})$, and thus $prices(s^{\mathcal{F}})[s^L] \geq Eprices(t^{\mathcal{F}})[s^L]$ on these states.

Note that, with the above, to evaluate $\preceq_E$ it suffices to compare the price of $s^{\mathcal{F}}$ vs. effective price of $t^{\mathcal{F}}$ on $F(s^{\mathcal{F}})$. This is equivalent to, but faster than, comparing all prices.

6 Simulation-Based Dominance

We use the concept of simulation relations [Milner, 1971; Gentilini *et al.*, 2003] *on leaf state spaces* in order to identify leaf states t^L which can do everything that another leaf state s^L can do.[3] In this situation, suppose that we are checking whether $s^{\mathcal{F}} \preceq t^{\mathcal{F}}$, and $prices(t^{\mathcal{F}})[s^L] > prices(s^{\mathcal{F}})[s^L]$, but $prices(t^{\mathcal{F}})[t^L] \leq prices(s^{\mathcal{F}})[s^L]$. Then $t^{\mathcal{F}}$ can still dominate $s^{\mathcal{F}}$, because if a solution for $s^{\mathcal{F}}$ relies on s^L, then starting from $t^{\mathcal{F}}$ we can use t^L instead. [4]

Definition 5 (Leaf simulation) *Let F^L be a leaf factor. A binary relation $\preceq^L$ on F^L leaf states is a* leaf simulation *if: $s^L_G \not\preceq^L s^L$ for all $s^L \neq s^L_G$; and whenever $s^L_1 \preceq^L t^L_1$, for every transition $s^L_1 \xrightarrow{a} s^L_2$ either (i) $s^L_2 \preceq^L t^L_1$ or (ii) there exists a transition $t^L_1 \xrightarrow{a'} t^L_2$ s.t. $s^L_2 \preceq^L t^L_2$, $\mathsf{pre}[F^C](a') \subseteq \mathsf{pre}[F^C](a)$, and $\mathsf{cost}(a') \leq \mathsf{cost}(a)$. We call $\preceq^L$ the* ***coarsest*** *leaf simulation if, for every leaf-simulation $\preceq'$, we have $\preceq' \subseteq \preceq^L$.*

This follows common notions, except for (i) which, intuitively, "allows t^L_1 to stay where it is", and except for allowing in (ii) different actions a' so long as they are at least as good in terms of center precondition and cost. The coarsest leaf simulation can be computed in time polynomial in the size of the leaf factor state space, as usual with simulation relations [Henzinger *et al.*, 1995], and it has similar properties, such as being reflexive and transitive.

It is easy to see that, whenever $s^L \preceq^L t^L$, if a leaf path π^L_s starting in s^L complies with a center path π^C, then there exists a π^C-compliant leaf path π^L_t starting in t^L s.t. $\mathsf{cost}(\pi^L_t) \leq \mathsf{cost}(\pi^L_s)$.

Lemma 1 *Let $\preceq^L$ be a leaf simulation and π^C a center path. If $s^L \preceq^L t^L$ and there exists a path π^L_s from s^L to s^L_G compliant with π^C, then there exists a path π^L_t from t^L to s^L_G compliant with π^C such that $\mathsf{cost}(\pi^L_t) \leq \mathsf{cost}(\pi^L_s)$.*

Consequently, we allow s^L to take a cheaper price from any leaf state that simulates it:

Definition 6 ($\preceq_S$ Relation) *The relation $\preceq_S$ over decoupled states is defined by $s^{\mathcal{F}} \preceq_S t^{\mathcal{F}}$ iff $cs(s^{\mathcal{F}}) = cs(t^{\mathcal{F}})$ and, for all $s^L \in S^L$, $prices(s^{\mathcal{F}})[s^L] \geq \min_{s^L \preceq^L t^L} prices(t^{\mathcal{F}})[t^L]$.*

Theorem 5 *$\preceq_S$ is a decoupled dominance relation.*

It is easy to see that this is strictly better than $\preceq_B$:

Theorem 6 *$\preceq_S$ subsumes $\preceq_B$ and is exponentially separated from it.*

[3] This is inspired by, but differs in scope and purpose from, the use of simulation relations on the state space for dominance pruning in standard search [Torralba and Hoffmann, 2015].

[4] Note that this is quite different from the recently proposed use of simulation relations on the state space for dominance pruning in standard search [Torralba and Hoffmann, 2015]. In that approach, the simulation *is* the dominance relation and the main difficulty is how to compute such a relation over the exponentially large state space. In our approach, the simulation relation is over small leaf state spaces, and it is a tool used *towards* defining a simulation relation over decoupled states.

The first part of this claim holds simply because $\preceq^L$ is reflexive (and therefore $\min_{s^L \preceq^L t^L} \mathit{prices}(t^{\mathcal{F}})[t^L] \leq \mathit{prices}(t^{\mathcal{F}})[s^L]$). For the second part, we use again our running example. Leaf simulation captures that $(p, l_i) \preceq^L (p, t)$ for all $i > 2$, since (p, t) is the only successor of any (p, l_i) and naturally $(p, t) \preceq^L (p, t)$. So, $\preceq_S$ reduces the price of such (p, l_i) to 1, avoiding the exponential blow-up.

Inspired by [Torralba and Kissmann, 2015], we also employ leaf simulation to discover superfluous leaf states and leaf actions. A transition $s^L \xrightarrow{a} t^L$ is superfluous if $t^L \preceq^L s^L$, or there exists another transition $s^L \xrightarrow{a'} u^L$ such that $t^L \preceq u^L$, $\mathsf{pre}[F^C](a') \subseteq \mathsf{pre}[F^C](a)$, and $\mathsf{cost}(a') \leq \mathsf{cost}(a)$. After removing such transitions, we run a reachability check on the leaf state space, removing unreachable leaf states. We subsequently remove leaf actions that do not induce any transition anymore. This reduces leaf state space size, and may sometimes improve the heuristic function due to the removal of some actions.

7 Method Interrelations and Combination

We have already established the relation of our methods relative to $\preceq_B$, as well as the relation between $\preceq_E$ and $\preceq_F$. We next design a combination $\preceq_{ES}$ of $\preceq_E$ and $\preceq_S$, with their respective strengths, and we establish the remaining method interrelations. Figure 1 provides the overall picture.

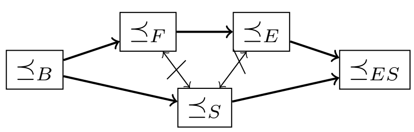


Figure 1: Summary of method interrelations. "$A \rightarrow B$": B subsumes A and is exponentially separated from it. "$A \not\leftrightarrow B$": A is exponentially separated from B and vice versa.

The combined relation $\preceq_{ES}$ is obtained by modifying the effective prices underlying $\preceq_E$, enriching their definition with a leaf simulation, $\preceq^L$. We define $\mathit{ESprices}(t^{\mathcal{F}})$ as the point-wise minimum pricing function p that satisfies:

$$p[s^L] = \begin{cases} \mathit{prices}(t^{\mathcal{F}})[s^L] & \text{if } s^L = s^L_G \\ \min\{\min_{s^L \preceq^L t^L} \mathit{prices}(t^{\mathcal{F}})[t^L], & \\ \quad \max_{s^L \xrightarrow{a} t^L} \big(p[t^L] - \mathsf{cost}(a)\big)\} & \text{otherwise} \end{cases}$$

We integrate the information from a leaf simulation into the effective prices by allowing s^L to take cheaper prices from simulating states t^L. This amounts to substituting $\mathit{prices}(t^{\mathcal{F}})[s^L]$ with $\min_{s^L \preceq^L t^L} \mathit{prices}(t^{\mathcal{F}})[t^L]$ in the equation. We thus obtain, again, a decoupled dominance relation:

Definition 7 (**$\preceq_{ES}$ Relation**) *$\preceq_{ES}$ is the relation over decoupled states defined by $s^{\mathcal{F}} \preceq_{ES} t^{\mathcal{F}}$ iff $cs(s^{\mathcal{F}}) = cs(t^{\mathcal{F}})$ and, for all $s^L \in S^L$, $\mathit{prices}(s^{\mathcal{F}})[s^L] \geq \mathit{ESprices}(t^{\mathcal{F}})[s^L]$.*

Theorem 7 *$\preceq_{ES}$ is a decoupled dominance relation.*

Theorem 7 is shown by adapting the property (*) underlying the proof of Theorem 3. Say $\pi^L_s = \langle a_1, \dots, a_n \rangle$ is a π^C-compliant goal leaf path starting in s^L, traversing the leaf states $s^L = s^L_0, \dots, s^L_n = s^L_G$. Then, with the same arguments as before, there exists i such that (a) $\mathit{ESprices}(t^{\mathcal{F}})[s^L_0] \geq \mathit{ESprices}(t^{\mathcal{F}})[s^L_i] - \sum_{j=1}^{i} \mathsf{cost}(a_i)$, and (b) $\mathit{ESprices}(t^{\mathcal{F}})[s^L_i] = \min_{s^L_i \preceq^L t^L} \mathit{prices}(t^{\mathcal{F}})[t^L]$. We construct our desired path π^L from s^L_I to s^L_G by a cheapest $cp(t^{\mathcal{F}})$-compliant path to a t^L minimizing the expression in (b), concatenated with a π^C-compliant goal leaf path π^L_t starting in t^L where $\mathsf{cost}(\pi^L_t) \leq \mathsf{cost}(\pi^L_s)$. Such π^L_t exists by the properties of leaf simulation, as in Theorem 5.

$\preceq_{ES}$ subsumes each of its components. The exponential separations therefore follow directly from the individual ones:

Theorem 8 *$\preceq_{ES}$ subsumes $\preceq_E$ and $\preceq_S$, and is exponentially separated from each of them.*

One can also construct cases where $\preceq_{ES}$ yields an exponentially stronger reduction than *both* $\preceq_E$ and $\preceq_S$, i. e., where $\preceq_{ES}$ is strictly more than the sum of its components. We complete our analysis by filling in the missing cases:

Theorem 9 *$\preceq_S$ is exponentially separated from $\preceq_E$, and therefore also from $\preceq_F$. $\preceq_F$, and therefore also $\preceq_E$, is exponentially separated from $\preceq_S$.*

8 Experiments

We implemented our dominance pruning methods within the fork-decoupled search variant of FD [Helmert, 2006] by GH. Our baseline is GH's basic pruning $\preceq_B$. For simplicity, we stick to the factoring strategy used by GH. This method greedily computes a factoring that maximizes the number of leaf factors. In case there are less than two leaves, the method *abstains* from solving a task. The rationale behind this is that the main advantage of decoupled search originates from not having to enumerate leaf state combinations across *multiple* leaf factors. Like GH, we show results on all IPC domains up to and including 2014 where the strategy does not abstain.

We focus on optimal planning, the main purpose of optimality-preserving pruning. We run a blind heuristic to identify the influence of different pruning methods per se, and we run LM-cut [Helmert and Domshlak, 2009] as a state-of-the-art heuristic. GH introduced two decoupled variants of A*, "Fork-Decoupled" A* and "Anytime Fork-Root" A*, which to simplify terminology we will refer to as *Decoupled A* (DA*)* and *Anytime Decoupled A* (ADA*)*. DA* is a direct application of A* to the decoupled state space. ADA* orders the open list based on the heuristic estimate of remaining center-cost, uses the heuristic estimate of remaining global-cost for pruning against the best solution so far, and runs until the open list is empty. Both algorithms result in similar coverage, with moderate differences in some domains. Our techniques turn out to be more beneficial for ADA*, which tends to have larger search spaces but less per-node runtime than DA*. We show detailed data for ADA*, and include data for baseline DA* (with $\preceq_B$) for comparison. All experiments are run on a cluster of Intel E5-2660 machines running at 2.20 GHz, with time (memory) cut-offs of 30 minutes (4 GB).

Table 1 shows the number of instances solved, comparing to both baselines DA* and ADA*. Data for DA* with the blind heuristic is not shown as it is identical to that for ADA*. The main gain for blind search stems from Miconic (+9), and NoMystery (+3). When using LM-cut, the advantage over

Expansions with Blind Heuristic: Improvement factor relative to $\preceq_B$

Domain	#	$\preceq_F$ $\sum$D	$\preceq_F$ GM	$\preceq_F$ max	$\preceq_E$ $\sum$D	$\preceq_E$ GM	$\preceq_E$ max	$\preceq_S$ $\sum$D	$\preceq_S$ GM	$\preceq_S$ max	$\preceq_{ES}$ $\sum$D	$\preceq_{ES}$ GM	$\preceq_{ES}$ max
Driverlog	11	1.0	1.0	1.0	5.0	1.8	6.5	2.4	1.3	2.8	5.0	1.8	6.5
Logistics00	22	1.2	1.0	1.2	2.5	1.4	3.8	2.5	1.4	3.8	2.5	1.4	3.8
Logistics98	4	1.0	1.0	1.0	3.9	2.1	4.2	2.3	1.7	2.4	3.9	2.1	4.2
Miconic	36	3.3	1.7	5.2	3.3	1.7	5.2	3.3	1.7	5.2	3.3	1.7	5.2
NoMystery	17	4.4	1.7	8.5	4.4	1.7	8.5	4.4	1.7	8.5	4.4	1.7	8.5
TPP	22	1.0	1.0	1.0	1.0	1.0	1.2	1.0	1.0	1.0	1.0	1.0	1.2
Zenotravel	11	1.0	1.0	1.0	1.4	1.1	1.6	1.3	1.1	1.5	1.4	1.1	1.6

Expansions with LM-cut: Improvement factor relative to $\preceq_B$

Domain	#	$\preceq_F$ $\sum$D	$\preceq_F$ GM	$\preceq_F$ max	$\preceq_E$ $\sum$D	$\preceq_E$ GM	$\preceq_E$ max	$\preceq_S$ $\sum$D	$\preceq_S$ GM	$\preceq_S$ max	$\preceq_{ES}$ $\sum$D	$\preceq_{ES}$ GM	$\preceq_{ES}$ max
Driverlog	13	1.0	1.0	1.0	2.4	1.3	4.3	1.9	1.2	3.4	2.4	1.3	4.3
Logistics00	25	1.0	1.0	1.0	2.1	1.2	2.3	1.4	1.3	3.0	2.2	1.4	3.0
Logistics98	6	1.0	1.0	1.0	1.7	1.3	1.7	109.8	10.2	1245.2	134.7	10.8	1245.2
Miconic	135	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
NoMystery	20	6.3	1.7	9.2	6.3	1.7	9.2	6.8	1.9	9.3	6.8	1.9	9.3
TPP	22	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Zenotravel	11	1.0	1.0	1.0	1.2	1.1	1.4	1.2	1.0	1.3	1.2	1.1	1.4

Runtime with Blind Heuristic: Improvement factor relative to $\preceq_B$

Domain	#	$\preceq_F$ $\sum$D	$\preceq_F$ GM	$\preceq_F$ max	$\preceq_E$ $\sum$D	$\preceq_E$ GM	$\preceq_E$ max	$\preceq_S$ $\sum$D	$\preceq_S$ GM	$\preceq_S$ max	$\preceq_{ES}$ $\sum$D	$\preceq_{ES}$ GM	$\preceq_{ES}$ max
Driverlog	9	0.9	0.9	1.0	30.7	2.6	38.9	10.3	2.2	14.4	35.3	2.9	47.5
Logistics00	7	1.4	1.3	1.5	6.4	5.9	15.2	8.4	8.3	22.5	7.5	7.0	19.7
Logistics98	3	0.8	0.8	0.8	21.2	4.1	22.4	12.1	5.4	12.3	26.4	6.2	27.5
Miconic	19	24.0	10.0	53.9	24.3	9.0	47.9	22.6	8.6	45.7	23.5	8.8	47.0
NoMystery	9	47.3	5.6	157.1	36.2	4.1	118.8	64.2	7.4	210.2	53.7	6.0	182.7
Pathways	2	0.9	0.9	0.9	0.7	0.7	0.7	1.0	1.0	1.0	0.6	0.6	0.6
Rovers	2	0.8	0.8	0.8	0.5	0.5	0.6	1.0	1.0	1.0	0.5	0.5	0.5
Satellite	3	0.9	0.9	1.0	0.6	0.7	0.9	1.0	1.0	1.0	0.5	0.6	0.8
TPP	13	0.8	0.8	1.0	0.0	0.1	0.3	0.1	0.3	0.8	0.0	0.1	0.3
Woodwork08	2	1.5	1.2	1.5	0.7	0.8	1.0	1.5	0.3	1.5	1.0	0.3	1.0
Woodwork11	1	1.5	1.5	1.5	0.7	0.7	0.7	1.5	1.5	1.5	1.0	1.0	1.0
Zenotravel	4	0.8	0.8	1.0	1.2	1.2	1.4	1.7	1.8	2.9	1.3	1.3	1.8

Runtime with LM-cut: Improvement factor relative to $\preceq_B$

Domain	#	$\preceq_F$ $\sum$D	$\preceq_F$ GM	$\preceq_F$ max	$\preceq_E$ $\sum$D	$\preceq_E$ GM	$\preceq_E$ max	$\preceq_S$ $\sum$D	$\preceq_S$ GM	$\preceq_S$ max	$\preceq_{ES}$ $\sum$D	$\preceq_{ES}$ GM	$\preceq_{ES}$ max
Driverlog	5	0.8	0.9	1.0	5.5	2.6	14.3	4.4	2.5	11.3	5.5	2.7	14.6
Logistics00	9	0.9	0.9	0.9	3.8	1.5	4.6	2.7	3.7	6.4	4.1	3.5	5.0
Logistics98	4	0.9	0.9	0.9	2.2	1.2	2.2	895.9	30.4	2643.9	750.2	26.2	2259.3
Miconic	81	0.9	1.0	1.2	1.0	0.9	1.1	1.0	1.0	1.2	0.9	0.9	1.0
NoMystery	12	13.3	3.0	21.0	12.6	2.9	22.4	16.2	3.8	28.9	14.6	3.6	26.0
Pathways	1	0.9	0.9	0.9	0.9	0.9	0.9	1.0	1.0	1.0	0.9	0.9	0.9
Rovers	5	0.9	0.9	0.9	0.7	0.7	0.8	1.0	1.0	1.0	0.7	0.7	0.8
Satellite	4	1.0	1.0	1.0	0.9	0.8	0.9	1.0	1.0	1.0	0.9	0.8	0.9
TPP	11	0.8	0.8	1.0	0.1	0.2	0.4	0.1	0.4	0.8	0.1	0.1	0.3
Woodwork08	8	1.0	1.0	1.1	1.0	1.0	1.0	1.2	0.9	1.7	1.1	0.8	1.4
Woodwork11	5	1.0	1.0	1.0	1.0	1.0	1.0	1.3	1.2	1.3	1.3	1.2	1.3
Zenotravel	4	0.9	0.9	1.0	1.1	1.0	1.2	1.3	1.3	1.6	1.1	1.1	1.3

Table 2: Improvement factor on commonly solved instances relative to $\preceq_B$, using ADA*. We show expansions up to last f-layer (top), and runtime (bottom), with the blind heuristic (left) and LM-cut (right). In the top part, some domains are skipped as all their factors are rounded to 1.0. In the bottom part, we only take into account the instances that are not trivially solved by all planners ($< 0.1s$). $\sum$D: Ratio over the per-domain sum. GM (max): geometric mean (maximum) of per-instance ratios.

		Blind Heuristic					LM-cut					
		ADA*					DA*	ADA*				
Domain	#	$\preceq_B$	$\preceq_F$	$\preceq_E$	$\preceq_S$	$\preceq_{ES}$	$\preceq_B$	$\preceq_B$	$\preceq_F$	$\preceq_E$	$\preceq_S$	$\preceq_{ES}$
Driverlog	20	11	11	11	11	11	13	13	13	13	13	13
Logistics00	28	22	22	22	22	22	**28**	25	25	27	26	**28**
Logistics98	35	4	4	**5**	**5**	**5**	6	6	6	6	6	6
Miconic	145	36	**45**	**45**	**45**	**45**	135	135	135	135	135	135
NoMystery	20	17	**20**	**20**	**20**	**20**	20	20	20	20	20	20
Pathways	29	3	3	3	3	3	4	4	4	4	4	4
Rovers	40	**7**	6	6	**7**	6	9	9	9	9	9	9
Satellite	36	**6**	**6**	**6**	**6**	5	7	**9**	**9**	8	**9**	**9**
TPP	27	**23**	**23**	22	**23**	22	18	**23**	**23**	22	22	22
Woodwork08	13	5	5	5	5	5	10	**11**	**11**	**11**	**11**	**11**
Woodwork11	5	1	1	1	1	1	4	**5**	**5**	**5**	**5**	**5**
Zenotravel	20	11	11	**12**	**12**	**12**	**13**	11	11	12	12	**13**
$\sum$	418	146	157	158	**160**	157	267	271	271	272	272	**275**

Table 1: Coverage data.

$\preceq_B$ is much smaller. We still gain +3 (+2) instances in Logistics00 (Zenotravel). In Satellite and TPP, we lose 1 instance in some configurations due to overhead at no search space reduction. $\preceq_{ES}$ reliably removes the disadvantages of ADA* relative to DA*, and is best in the overall. We never strictly improve coverage over both baselines, though. As we shall see below, this is due to benchmark scaling, i. e., there *are* domains where runtime is improved over both baselines.

We next analyze the search space size reduction (top part of Table 2). In general, the blind heuristic has more margin of improvement except in Logistics98, where the improvement with LM-cut gets magnified due to the relevance analysis performed when enabling $\preceq_S$. In that domain, removing irrelevant leaf states and leaf actions renders LM-cut a lot stronger.[5] Regarding the relative behavior of pruning techniques, in two domains, namely Miconic and NoMystery, already the simplest technique ($\preceq_F$) gets the maximal improvement factor. In four domains, enabling effective-price pruning on top of frontier pruning results in additional pruning. Combining all techniques in $\preceq_{ES}$ always inherits the strongest search space reduction of its components and in Logistics with LM-cut, it often is strictly better.

Consider now runtime, Table 2 bottom. One key observation is that, whenever the search space is reduced, the same holds for runtime, even for small search space reduction factors like, e. g., in Zenotravel. Remarkably, in some domains (e. g. Woodworking) where no search reduction is obtained, runtime decreases nevertheless for some simple methods such as $\preceq_F$. This is due to the cheaper dominance check – prices are compared only on *frontier* leaf states. There are also some bad cases, though, mainly in TPP, but also in Pathways, Rovers, and Satellite. These are also the domains in which coverage slightly decreases. What makes these domains special is the structure of their leaf state spaces. In Pathways, Rovers, and Satellite, all leaves are single variables with a single transition, $s^L_I \rightarrow s^L_G$, so there is no room for improvement. In TPP, the leaf state spaces are quite large (up to 5000 states), so our methods incur substantial overhead, but are unable to perform pruning. Presumably, this is because most of the leaf states can play a role in optimally reaching the goal.

Coming back to our previous observation that coverage is never improved over both baselines, the runtime analysis reveals an improvement over both baselines in several domains. ADA* with $\preceq_S$ is faster than DA* with $\preceq_B$ in all domains except Zenotravel, where the geomean per-instance runtime factor is 0.7. The other factors are: Driverlog 2.3; Logistics00 2.3; Logistics98 3.4; Miconic 2.7; NoMystery 3.2; Pathways

[5] It may be surprising that, elsewhere, the improvements in Logistics are moderate, despite the inherent blow-up we explained earlier. This is because, in the commonly solved instances, the number of non-airport locations in each city is very small, mostly 1.

1.1; Rovers 2.1; Satellite 2.9; TPP 23.2; Woodworking08 1.4; and Woodworking11 2.0. In particular, in Driverlog, both Logistics domains, NoMystery, and Woodworking11, ADA* with $\preceq_S$ improves runtime over both baselines.

Finally, consider the use of our pruning methods in DA*. For blind search, the numbers are almost identical to those for ADA* in Table 2, as DA* and ADA* differ mainly in their use of a (non-trivial) heuristic. With LM-cut, the pruning methods do not work as well for DA*. For example, for $\preceq_S$, the geomean per-instance runtime factors are: Driverlog 1.8; Logistics00 and Logistics98 2.5; NoMystery 2.0; TPP 0.9; Woodworking08 0.9; Woodworking11 1.3; Zenotravel 1.2; and 1.0 in the other domains. The picture is similar for the other pruning methods. The big runtime advantages observed with ADA* vanish, but the method also becomes less risky, i. e., the big runtime disadvantage in TPP vanishes as well. This makes sense since DA* searches less nodes (it has less potential for pruning) while spending more time on each node (making the dominance-checking overhead less pronounced).

9 Conclusion

Dominance pruning methods can be quite useful for decoupled search. Our analysis of such methods is fairly complete, although of course other variants may be thinkable. More pressingly, the question remains whether there exist duplicate checking methods guaranteeing to avoid all blow-ups.

Acknowledgments

This work was partially supported by the German Research Foundation (DFG), under grant HO 2169/6-1, “Star-Topology Decoupled State Space Search”.

References

[Aghighi *et al.*, 2015] Meysam Aghighi, Peter Jonsson, and Simon Ståhlberg. Tractable cost-optimal planning over restricted polytree causal graphs. In Blai Bonet and Sven Koenig, editors, *Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI'15)*, pages 3225–3231. AAAI Press, January 2015.

[Amir and Engelhardt, 2003] Eyal Amir and Barbara Engelhardt. Factored planning. In G. Gottlob, editor, *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI'03)*, pages 929–935, Acapulco, Mexico, August 2003. Morgan Kaufmann.

[Bäckström and Nebel, 1995] Christer Bäckström and Bernhard Nebel. Complexity results for SAS$^+$ planning. *Computational Intelligence*, 11(4):625–655, 1995.

[Brafman and Domshlak, 2003] Ronen Brafman and Carmel Domshlak. Structure and complexity in planning with unary operators. *Journal of Artificial Intelligence Research*, 18:315–349, 2003.

[Brafman and Domshlak, 2013] Ronen Brafman and Carmel Domshlak. On the complexity of planning for agent teams and its implications for single agent planning. *Artificial Intelligence*, 198:52–71, 2013.

[Fabre *et al.*, 2010] Eric Fabre, Loïg Jezequel, Patrik Haslum, and Sylvie Thiébaux. Cost-optimal factored planning: Promises and pitfalls. In Ronen I. Brafman, Hector Geffner, Jörg Hoffmann, and Henry A. Kautz, editors, *Proceedings of the 20th International Conference on Automated Planning and Scheduling (ICAPS'10)*, pages 65–72. AAAI Press, 2010.

[Gentilini *et al.*, 2003] Raffaella Gentilini, Carla Piazza, and Alberto Policriti. From bisimulation to simulation: Coarsest partition problems. *Journal of Automated Reasoning*, 31(1):73–103, 2003.

[Gnad and Hoffmann, 2015] Daniel Gnad and Jörg Hoffmann. Beating LM-cut with h^{max} (sometimes): Fork-decoupled state space search. In Ronen Brafman, Carmel Domshlak, Patrik Haslum, and Shlomo Zilberstein, editors, *Proceedings of the 25th International Conference on Automated Planning and Scheduling (ICAPS'15)*. AAAI Press, 2015.

[Helmert and Domshlak, 2009] Malte Helmert and Carmel Domshlak. Landmarks, critical paths and abstractions: What's the difference anyway? In Alfonso Gerevini, Adele Howe, Amedeo Cesta, and Ioannis Refanidis, editors, *Proceedings of the 19th International Conference on Automated Planning and Scheduling (ICAPS'09)*, pages 162–169. AAAI Press, 2009.

[Helmert, 2006] Malte Helmert. The Fast Downward planning system. *Journal of Artificial Intelligence Research*, 26:191–246, 2006.

[Henzinger *et al.*, 1995] Monika Rauch Henzinger, Thomas A. Henzinger, and Peter W. Kopke. Computing simulations on finite and infinite graphs. In *Proceedings of the 36th Annual Symposium on Foundations of Computer Science (FOCS'95)*, pages 453–462. IEEE Computer Society, 1995.

[Jonsson and Bäckström, 1995] Peter Jonsson and Christer Bäckström. Incremental planning. In *European Workshop on Planning*, 1995.

[Katz and Domshlak, 2008] Michael Katz and Carmel Domshlak. Structural patterns heuristics via fork decomposition. In Jussi Rintanen, Bernhard Nebel, J. Christopher Beck, and Eric Hansen, editors, *Proceedings of the 18th International Conference on Automated Planning and Scheduling (ICAPS'08)*, pages 182–189. AAAI Press, 2008.

[Katz and Keyder, 2012] Michael Katz and Emil Keyder. Structural patterns beyond forks: Extending the complexity boundaries of classical planning. In Jörg Hoffmann and Bart Selman, editors, *Proceedings of the 26th AAAI Conference on Artificial Intelligence (AAAI'12)*, pages 1779–1785, Toronto, ON, Canada, July 2012. AAAI Press.

[Kelareva *et al.*, 2007] Elena Kelareva, Olivier Buffet, Jinbo Huang, and Sylvie Thiébaux. Factored planning using decomposition trees. In M. Veloso, editor, *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI'07)*, pages 1942–1947, Hyderabad, India, January 2007. Morgan Kaufmann.

[Knoblock, 1994] Craig Knoblock. Automatically generating abstractions for planning. *Artificial Intelligence*, 68(2):243–302, 1994.

[Milner, 1971] Robin Milner. An algebraic definition of simulation between programs. In *Proceedings of the 2nd International Joint Conference on Artificial Intelligence (IJCAI'71)*, pages 481–489, London, UK, September 1971. William Kaufmann.

[Torralba and Hoffmann, 2015] Álvaro Torralba and Jörg Hoffmann. Simulation-based admissible dominance pruning. In Qiang Yang, editor, *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI'15)*, pages 1689–1695. AAAI Press/IJCAI, 2015.

[Torralba and Kissmann, 2015] Álvaro Torralba and Peter Kissmann. Focusing on what really matters: Irrelevance pruning in merge-and-shrink. In Levi Lelis and Roni Stern, editors, *Proceedings of the 8th Annual Symposium on Combinatorial Search (SOCS'15)*, pages 122–130. AAAI Press, 2015.

A Proofs

Theorem 1 *$\preceq_F$ is a decoupled dominance relation.*

Proof: Whenever $s^{\mathcal{F}} \preceq_F t^{\mathcal{F}}$ and a completion plan for $s^{\mathcal{F}}$ exists, we can construct an at least as good completion plan for $t^{\mathcal{F}}$. Consider any completion plan for $s^{\mathcal{F}}$ consisting of a center path π^C, and a goal leaf path π^L for each leaf starting in s^L_I compliant with $cp(s^{\mathcal{F}}) \circ \pi^C$. As $s^{\mathcal{F}} \preceq_F t^{\mathcal{F}}$ implies that $cs(s^{\mathcal{F}}) = cs(t^{\mathcal{F}})$, π^C is applicable to $t^{\mathcal{F}}$ as well.

Denote $\pi^L = \langle a_1, \dots, a_n \rangle$ and denote the leaf states it traverses by $s^L_I = s^L_0, \dots, s^L_n = s^L_G$. Let s^L_i be the last state visited by π^L such that $prices(s^{\mathcal{F}})[s^L_i] = \sum_{j=1}^{i} \mathsf{cost}(a_j)$. Such state must exist because the price of the initial state is always 0, $prices(s^{\mathcal{F}})[s^L_0] = 0$. s^L_i is the last state in which its price corresponds to the price obtained by π^L so $\langle a_{i+1}, \dots, a_n \rangle$ must be compliant with π^C.

It follows that $s^L_i \in F(s^{\mathcal{F}})$. This is trivially true if $s^L_i = s^L_n$ since the goal state always belongs to the frontier. If $s^L_i \neq s^L_n$ then $prices(s^{\mathcal{F}})[s^L_i] + \mathsf{cost}(a_{i+1}) < prices(s^{\mathcal{F}})[s^L_{i+1}]$, which implies $s^L_i \in F(s^{\mathcal{F}})$. Since $s^{\mathcal{F}} \preceq_F t^{\mathcal{F}}$, it follows that $prices(t^{\mathcal{F}})[s^L_i] \leq prices(s^{\mathcal{F}})[s^L_i]$.

Consider the path π^L_t from s^L_I to s^L_G constructed as the concatenation of: a cheapest $cp(t^{\mathcal{F}})$-compliant path to s^L_i with the postfix of π^L behind s^L_i. Then $\mathsf{cost}(\pi^L_t) = prices(t^{\mathcal{F}})[s^L_i] + \sum_{j=i+1}^{n} \mathsf{cost}(a_j) \leq \mathsf{cost}(\pi^L)$. Since the first part of the plan up to s^L_i is compliant with $cp(t^{\mathcal{F}})$ and the rest is compliant with π^C, π^L_t is compliant with $cp(t^{\mathcal{F}}) \circ \pi^C$ as desired. □

Theorem 2 *$\preceq_F$ subsumes $\preceq_B$ and is exponentially separated from it.*

Proof: Subsumption is guaranteed as both relations are based on comparing the same pricing functions, but $\preceq_F$ does so on a subset of leaf states. The exponential separation is shown by our running example, in which decoupled search with $\preceq_B$ incurs an exponential blow-up, as explained before.

Note that the price of (p, l_1) and (p, t) are always 0 and 1, respectively. Since the only leaf action applicable from (p, l_i) is $load(p, l_i)$ and the price of (p, t) is always lower than that of (p, l_i) for $i > 1$, $(p, l_i) \notin F(s^{\mathcal{F}}), i > 2$ for any $s^{\mathcal{F}}$. Hence, when using $\preceq_F$, there are at most 2 different decoupled states for every center state, depending on whether the price of (p, l_2) is 2 or ∞. Therefore, the decoupled state space with $\preceq_F$ has size linear in n instead of exponential. □

Theorem 3 *$\preceq_E$ is a decoupled dominance relation.*

Proof: Whenever $s^{\mathcal{F}} \preceq_E t^{\mathcal{F}}$ and a completion plan for $s^{\mathcal{F}}$ exists, we can construct an at least as good completion plan for $t^{\mathcal{F}}$. Consider any completion plan for $s^{\mathcal{F}}$ consisting of a center path π^C, and a goal leaf path π^L for each leaf starting in s^L_I compliant with $cp(s^{\mathcal{F}}) \circ \pi^C$. As $s^{\mathcal{F}} \preceq_E t^{\mathcal{F}}$ implies that $cs(s^{\mathcal{F}}) = cs(t^{\mathcal{F}})$, π^C is applicable to $t^{\mathcal{F}}$ as well. Let s^L be a leaf state traversed by π^L such that the prefix of π^L up to s^L is compliant with $cp(s^{\mathcal{F}})$ and the suffix of π^L starting in s^L, π^L_s, is compliant with π^C. Denote $\pi^L_s = \langle a_1, \dots, a_n \rangle$ and denote the leaf states it traverses by $s^L = s^L_0, \dots, s^L_n = s^L_G$.

Let s^L_i be the first state visited by π^L_s such that $Eprices(t^{\mathcal{F}})[s^L_i] = prices(t^{\mathcal{F}})[s^L_i]$. Such state exists by definition since $Eprices(t^{\mathcal{F}})[s^L_n] = prices(t^{\mathcal{F}})[s^L_n]$. Then, for all $j < i$, $Eprices(t^{\mathcal{F}})[s^L_j] \neq prices(t^{\mathcal{F}})[s^L_j]$, and thus by the definition of effective prices we have that $Eprices(t^{\mathcal{F}})[s^L_j] \geq Eprices(t^{\mathcal{F}})[s^L_{j+1}] - \mathsf{cost}(a_{j+1})$. Accumulating these inequalities, we get (*) $Eprices(t^{\mathcal{F}})[s^L_0] \geq Eprices(t^{\mathcal{F}})[s^L_i] - \sum_{j=1}^{i} \mathsf{cost}(a_j)$.

Consider now the path π^L_t from s^L_I to s^L_G constructed as the concatenation of: a cheapest $cp(t^{\mathcal{F}})$-compliant path to s^L_i with the postfix of π^L_s behind s^L_i. Then $\mathsf{cost}(\pi^L_t) = prices(t^{\mathcal{F}})[s^L_i] + \sum_{j=i+1}^{n} \mathsf{cost}(a_j)$. As $Eprices(t^{\mathcal{F}})[s^L_i] = prices(t^{\mathcal{F}})[s^L_i]$, we get $\mathsf{cost}(\pi^L_t) = Eprices(t^{\mathcal{F}})[s^L_i] + \sum_{j=i+1}^{n} \mathsf{cost}(a_j)$. With (*), we get the desired property that:

$$
\begin{aligned}
& \mathsf{cost}(\pi^L_t) \\
\leq\; & Eprices(t^{\mathcal{F}})[s^L] + \textstyle\sum_{j=1}^{i} \mathsf{cost}(a_j) + \sum_{j=i+1}^{n} \mathsf{cost}(a_j) \\
=\; & Eprices(t^{\mathcal{F}})[s^L] + \mathsf{cost}(\pi^L_s) \\
\leq\; & prices(s^{\mathcal{F}})[s^L] + \mathsf{cost}(\pi^L_s) \\
=\; & \mathsf{cost}(\pi^L)
\end{aligned}
$$

□

Theorem 4 *$\preceq_E$ subsumes $\preceq_F$ and is exponentially separated from it.*

Proof: To prove that $\preceq_E$ subsumes $\preceq_F$, we show that if $Eprices(t^{\mathcal{F}})[s^L] \leq prices(s^{\mathcal{F}})[s^L]$ for all $s^L \in F(s^{\mathcal{F}})$, then it also holds for all $s^L \notin F(s^{\mathcal{F}})$.

Consider any $s^L \notin F(s^{\mathcal{F}})$. Let $R^L \subseteq F(s^{\mathcal{F}})$ be the set of leaf states in the frontier of $s^{\mathcal{F}}$ reachable from s^L through a path that does not traverse any other state in $F(s^{\mathcal{F}})$. Note that, since $s^L_G \in F(s^{\mathcal{F}})$, any path from s^L to s^L_G necessarily passes through some $r^L \in R^L$. Let r^L be $\operatorname{argmax}_{t^L \in R^L} Eprices(t^{\mathcal{F}})[t^L] - \mathsf{cost}(s^L, t^L)$ where $\mathsf{cost}(s^L, t^L)$ is the cost of the optimal path from s^L to t^L.

First we show that $Eprices(t^{\mathcal{F}})[s^L] \leq Eprices(t^{\mathcal{F}})[r^L] - \mathsf{cost}(s^L, r^L)$. By definition, $Eprices(t^{\mathcal{F}})[s^L] \leq \max_{s^L \xrightarrow{a} t^L}(Eprices(t^{\mathcal{F}})[t^L] - \mathsf{cost}(a))$. Let s_1^L be any t^L that maximizes such expression. If $s_1^L \in R^L$ then $Eprices(t^{\mathcal{F}})[s^L] \leq Eprices(t^{\mathcal{F}})[s_1^L] - \mathsf{cost}(s^L, s_1^L) \leq Eprices(t^{\mathcal{F}})[r^L] - \mathsf{cost}(s^L, r^L)$ and we are done. If $s_1^L \notin R^L$ then we can compose the definition to obtain, $Eprices(t^{\mathcal{F}})[s^L] \leq Eprices(t^{\mathcal{F}})[s_1^L] - \mathsf{cost}(a) \leq \max_{s_1^L \xrightarrow{a} s_2^L}(Eprices(t^{\mathcal{F}})[s_2^L] - \mathsf{cost}(a))$. By repeating the same argument we have that at least one state $u^L \in R^L$. This is clear if there are no 0-cost actions because $Eprices(t^{\mathcal{F}})[s_i^L] < Eprices(t^{\mathcal{F}})[s_{i+1}^L]$ so the $Eprices(t^{\mathcal{F}})$ monotonically increases during this chain and there can be no cycles.

If there are 0-cost actions, there may be a 0-cost cycle, so we must prove that even in that case, there exists yet another state u^L outside the cycle such that $Eprices(t^{\mathcal{F}})[s^L] \leq Eprices(t^{\mathcal{F}})[u^L] - \mathsf{cost}(s^L, u^L)$. Consider the set of states, S_s^L for which there is a 0-cost cycle such that $S_s^L \cap R^L = \emptyset$ and all the states in S_s^L have the same $Eprices(t^{\mathcal{F}})$. Let C be a constant such that $Eprices(t^{\mathcal{F}})[s_i^L] = C$ for all $s_i^L \in S_s^L$. Then, there exists $t^L \notin S_s^L$ such that $s_i^L \xrightarrow{a} t^L$, $Eprices(t^{\mathcal{F}})[s_i^L] \leq Eprices(t^{\mathcal{F}})[t^L] - \mathsf{cost}(a)$ for some $s_i^L \in S_s^L$. Otherwise, $Eprices(t^{\mathcal{F}})[s_i^L] = C'$ for all $s_i^L \in S_i^L$ and some $C' < C$ would satisfy the equation for which $Eprices$ is defined to be the minimum point-wise that satisfies it.

Finally, since $s^L \notin F(s^{\mathcal{F}})$, $prices(s^{\mathcal{F}})[s^L] \geq prices(s^{\mathcal{F}})[t^L] - \mathsf{cost}(a)$ for all $s^L \xrightarrow{a} t^L$. Since all states in the path from s^L to any $r^L \in R^L$ do not belong to $F(s^{\mathcal{F}})$, we have the inequality $prices(s^{\mathcal{F}})[s^L] \geq prices(s^{\mathcal{F}})[r^L] - \mathsf{cost}(s^L, r^L)$. Combining the previous inequalities we get:

$$\begin{aligned} & Eprices(t^{\mathcal{F}})[s^L] \\ \leq \; & Eprices(t^{\mathcal{F}})[r^L] - \mathsf{cost}(s^L, r^L) \\ \leq \; & prices(s^{\mathcal{F}})[r^L] - \mathsf{cost}(s^L, r^L) \\ \leq \; & prices(s^{\mathcal{F}})[s^L] \end{aligned}$$

For the exponential separation, consider the planning task of our running example with an additional teleport device that allows to teleport the package between any two locations that are not the goal. Hence, we have a new center variable *tel-activated* with domain $\{F, T\}$, initially set to F and two new actions *activate-teleport* that sets *tel-activated* to T and *teleport-package*(l_i, l_j) for $i, j \neq 2$ that has (*tel-activated*, T) *and* (p, l_i) *as preconditions and* (p, l_j) *as effect. The cost of activate-teleport is* n *and all other actions have cost* 1.

In this example, decoupled search with $\preceq_F$ *expands an exponential number of states with* (*tel-activated*, F) *since for all reached* (p, l_i) *with* $i > 2$, $(p, l_i) \in F(s^{\mathcal{F}})$ *as long as any* (p, l_i) *is still not reached. When using* $\preceq_E$, $prices(I^{\mathcal{F}})[(p, t)] = 1$ *in every state, so* $Eprices(s^{\mathcal{F}})[(p, l_i)] = 0$. *Therefore, the state space is linear in the number of locations.*

□

Lemma 1 *Let* $\preceq^L$ *be a leaf simulation and* π^C *a center path. If* $s^L \preceq^L t^L$ *and there exists a path* π_s^L *from* s^L *to* s_G^L *compliant with* π^C, *then there exists a path* π_t^L *from* t^L *to* s_G^L *compliant with* π^C *such that* $\mathsf{cost}(\pi_t^L) \leq \mathsf{cost}(\pi_s^L)$.

Proof: Assume without loss of generality that π_s^L is a shortest optimal path compliant with π^C. Proof by induction on $|\pi_s^L|$. Base case, $|\pi_s^L| = 0$. Then, $s^L = s_G^L$ and $t^L = s_G^L$ because $s_G^L \not\preceq^L t^L$ for any $t^L \neq s_G^L$. Thus, $\pi_s^L = \pi_t^L = \langle\rangle$.

Inductive case. Let $s^L \xrightarrow{a_s^L} s_2^L$ be the first transition in π_s^L. Since $s^L \preceq^L t^L$, either (i) $s_2^L \preceq^L t^L$ or (ii) exists $t^L \xrightarrow{a_t^L} t_2^L$ such that a_t^L dominates a_s^L and $s_2^L \preceq t_2^L$. If (i) holds then the claim follows by induction since it must exist π_t^L with $\mathsf{cost}(\pi_t^L) \leq \mathsf{cost}(\pi_{s2}^L)$ where π_{s2}^L is a shortest optimal π^C-compliant plan from s_2^L and necessarily $\mathsf{cost}(\pi_{s2}^L) \leq \mathsf{cost}(\pi_s^L)$ and π_{s2}^L is shorter than π_s^L.

If (ii) holds, by induction we know that the claim holds for s_2^L and t_2^L, so exists a plan π_{t2}^L. Thus, $\mathsf{cost}(\pi_t^L) = \mathsf{cost}(\pi_{t2}^L) + c(a_t^L) \leq c(\pi_{s2}^L) + c(a_t^L) \leq \mathsf{cost}(\pi_{s2}^L) + c(a_s^L) = c(\pi_s^L)$. Moreover, since π_s^L is compliant with π^C and $\mathsf{pre}^C(a_t^L) \subseteq \mathsf{pre}^C(a_s^L)$, π_t^L is also compliant with π^C. □

Theorem 5 $\preceq_S$ *is a decoupled dominance relation.*

Proof: Whenever $s^{\mathcal{F}} \preceq_S t^{\mathcal{F}}$ and a completion plan for $s^{\mathcal{F}}$ exists, we can construct an at least as good completion plan for $t^{\mathcal{F}}$. Consider any completion plan for $s^{\mathcal{F}}$ consisting of a center path π^C, and a goal leaf path π^L for each leaf starting in s_I^L compliant with $cp(s^{\mathcal{F}}) \circ \pi^C$. As $s^{\mathcal{F}} \preceq_S t^{\mathcal{F}}$ implies that $cs(s^{\mathcal{F}}) = cs(t^{\mathcal{F}})$, π^C is applicable to $t^{\mathcal{F}}$ as well. Let s^L be a leaf state traversed by π^L such that the prefix of π^L up to s^L is compliant with $cp(s^{\mathcal{F}})$ and the suffix of π^L starting in s^L, π_s^L, is compliant with π^C. Denote $\pi_s^L = \langle a_1, \dots, a_n \rangle$ and denote the leaf states it traverses by $s^L = s_0^L, \dots, s_n^L = s_G^L$.

Since $s^{\mathcal{F}} \preceq_S t^{\mathcal{F}}$, $\min_{s^L \preceq^L t^L} prices(t^{\mathcal{F}})[t^L] \leq prices(s^{\mathcal{F}})[s^L]$. Then, exists t^L such that $s^L \preceq^L t^L$ and $prices(t^{\mathcal{F}})[t^L] \leq prices(s^{\mathcal{F}})[s^L]$. By Lemma 1, exists a π^C-compliant leaf plan π_t^L from t^L to s_G^L such that $\mathsf{cost}(\pi_t^L) \leq \mathsf{cost}(\pi_s^L)$. Consider now the path π_2^L from s_I^L to s_G^L constructed as the concatenation of: a cheapest $cp(t^{\mathcal{F}})$-compliant path to t^L and π_t^L. By definition, π_2^L is compliant with $cp(t^{\mathcal{F}}) \circ \pi^C$. Then, $\mathsf{cost}(\pi_2^L) = prices(t^{\mathcal{F}})[t^L] + \mathsf{cost}(\pi_t^L) \leq prices(s^{\mathcal{F}})[s^L] + \mathsf{cost}(\pi_s^L) = \mathsf{cost}(\pi^L)$.

□

Theorem 6 $\preceq_S$ *subsumes* $\preceq_B$ *and is exponentially separated from it.*

Proof: The subsumption holds simply because $\preceq^L$ is reflexive, so $\min_{s^L \preceq^L t^L} prices(t^{\mathcal{F}})[t^L] \leq prices(t^{\mathcal{F}})[s^L]$.

To show the exponential separation, we use again our running example. Leaf simulation captures that $(p, l_i) \preceq^L (p, t)$ for all $i \neq 2$. As $prices(I^{\mathcal{F}})[(p, t)] = 1$, $prices(s^{\mathcal{F}})[(p, l_i)] = 1$ for $i > 2$ in any decoupled state. Therefore, the size of the decoupled state space when using $\preceq_S$ is linear in the number of locations.

□

Theorem 7 $\preceq_{ES}$ *is a decoupled dominance relation.*

Proof: Whenever $s^{\mathcal{F}} \preceq_{ES} t^{\mathcal{F}}$ and a completion plan for $s^{\mathcal{F}}$ exists, we can construct an at least as good completion plan for $t^{\mathcal{F}}$. Consider any completion plan for $s^{\mathcal{F}}$ consisting of a center path π^C, and a goal leaf path π^L for each leaf starting in s^L_I compliant with $cp(s^{\mathcal{F}}) \circ \pi^C$. As $s^{\mathcal{F}} \preceq_{ES} t^{\mathcal{F}}$ implies that $cs(s^{\mathcal{F}}) = cs(t^{\mathcal{F}})$, π^C is applicable to $t^{\mathcal{F}}$ as well. Let s^L be a leaf state traversed by π^L such that the prefix of π^L up to s^L is compliant with $cp(s^{\mathcal{F}})$ and the suffix of π^L starting in s^L, π^L_s, is compliant with π^C. Denote $\pi^L_s = \langle a_1, \dots, a_n \rangle$ and denote the leaf states it traverses by $s^L = s^L_0, \dots, s^L_n = s^L_G$.

By the same arguments as in the proof of Theorem 3, there exists i such that (a) $ESprices(t^{\mathcal{F}})[s^L_0] \geq ESprices(t^{\mathcal{F}})[s^L_i] - \sum_{j=1}^{i} \mathsf{cost}(a_j)$, and (b) $ESprices(t^{\mathcal{F}})[s^L_i] = \min_{s^L_i \preceq^L t^L} prices(t^{\mathcal{F}})[t^L]$.

Hence, there exists t^L such that $prices(t^{\mathcal{F}})[t^L] \leq prices(s^{\mathcal{F}})[s^L] + \sum_{j=1}^{i} \mathsf{cost}(a_j)$. Also, by Lemma 1, exists a π^C-compliant leaf plan, π^L_t, from t^L to s^L_G such that $\mathsf{cost}(\pi^L_t) \leq \sum_{j=i+1}^{n} \mathsf{cost}(a_j)$.

We construct our desired path π^L_2 from s^L_I to s^L_G by a cheapest $cp(t^{\mathcal{F}})$-compliant path to t^L, concatenated with π^L_t. In summary:

$$\begin{aligned} & \mathsf{cost}(\pi^L_2) \\ = \; & prices(t^{\mathcal{F}})[t^L] + \mathsf{cost}(\pi^L_t) \\ \leq \; & prices(s^{\mathcal{F}})[s^L_i] + \mathsf{cost}(\pi^L_t) \\ \leq \; & prices(s^{\mathcal{F}})[s^L] + \textstyle\sum_{j=1}^{i} \mathsf{cost}(a_j) + \mathsf{cost}(\pi^L_t) \\ \leq \; & prices(s^{\mathcal{F}})[s^L] + \mathsf{cost}(\pi^L_s) \\ = \; & \mathsf{cost}(\pi^L) \end{aligned}$$

□

Theorem 8 *$\preceq_{ES}$ subsumes $\preceq_E$ and $\preceq_S$, and is exponentially separated from each of them.*

Proof: By definition, $ESprices(s^{\mathcal{F}})[s^L] \leq Eprices(s^{\mathcal{F}})[s^L]$ and $ESprices(s^{\mathcal{F}})[s^L] \leq \min_{s^L \preceq^L t^L} prices(t^{\mathcal{F}})[t^L]$ so subsumption follows.

To prove the exponential separation, consider the following planning task, in which we have a single center variable c (see Figure 2a) with domain c_0, c_{n+1}, and c^A_i, c^B_i for $i \in [1, \dots, n]$, arranged such that we can move from c_0 to c^X_1, from c^X_n to c_{n+1}, and from each c^X_i to each c^Y_{i+1} ($X, Y \in \{A, B\}$). The initial value is c_0. We have a single leaf variable l with values l_0, l_{n+1}, and l^A_i, l^B_i, l^{A2}_i, l^{B2}_i for $i \in [1, \dots, n]$ with initial value l_0 and goal value l_{n+1}. The transitions of the leaf l are depicted in Figure 2b. l has transitions from l_0 to l^X_1 under center precondition c^X_1, from l^{X2}_n to l_{n+1} under center precondition c_{n+1}, from each l^X_i to l^{X2}_i under center precondition c^X_i, and from each l^{X2}_i to each l^X_{i+1} and l^Y_{i+1} under center precondition c^X_i and c^Y_i, respectively ($X, Y \in \{A, B\}, X \neq Y$).

In this example, each of the separated techniques expands an exponential number of states in n:

- With $\preceq_B$, at each step in the center path, one of the two paths, A or B is chosen. Hence, the pricing function "remembers" which one of the two paths were taken and there is an exponential number of combinations.
- $\preceq_F$ and $\preceq_E$ ignore the prices of l^X_i because they do not belong to the frontier. However, they still remember whether l^{A2}_i or l^{B2}_i has been reached. $\preceq_E$ helps whenever l_n is reached but those states have the largest g-value (equal to $n+1$) so cannot prune any other state.
- $\preceq_S$ finds the following leaf-simulation relation:

  $$\begin{aligned} \preceq^L = & \{(l^X_i, l^{Y2}_j), (l^{X2}_i, l^{Y2}_j) \text{ for all } i \leq j, X, Y \in A, B\} \\ & \cup \{(x, x), (x, l_n) \text{ for all } x \in S^L\} \end{aligned}$$

  In summary, according to $\preceq^L$ states of the form l^{A2}_i, l^{B2}_i are equivalent and they simulate all previous states. However, nothing is found for states l^A_i and l^B_i.

  Therefore, decoupled search with $\preceq_S$ expands the same states than with $\preceq_B$. The main reason is that, even though it finds the equivalence between l^{A2}_i and l^{B2}_i, depending which path was used, either l^A_i or l^B_i will have a lower price.

  For example, consider the states $s^{\mathcal{F}}$ and $t^{\mathcal{F}}$ reached from the initial decoupled state by applying the center actions $c_0 \to c^A_1$ and $c_0 \to c^B_1$, respectively. Then, $prices[s^{\mathcal{F}}](l^A_1) = 1$ and $prices[s^{\mathcal{F}}](l^{A2}_1) = 2$, $prices[t^{\mathcal{F}}](l^B_1) = 1$ and $prices[t^{\mathcal{F}}](l^{B2}_1) = 2$. The initial state has price 0 and other leaf states ∞. In this case, $s \not\preceq_S t$ (nor $t \not\preceq_S s$) because $prices(s^{\mathcal{F}})[l^A_1] = 1 < \min_{s^L \preceq^L t^L} prices(t^{\mathcal{F}})[t^L] = 2$.

Finally, by using $\preceq_{ES}$, the decoupled state space is polynomial in n. Since $l^{A2}_i \preceq^L l^{B2}_i$ and $l^{B2}_i \preceq^L l^{A2}_i$, $ESprices(t^{\mathcal{F}})[l^{A2}_i] = ESprices(t^{\mathcal{F}})[l^{B2}_i]$. Hence, since the only transitions from l^A_i and l^B_i go to l^{A2}_i and l^{B2}_i, $ESprices(t^{\mathcal{F}})[l^A_i] = ESprices(t^{\mathcal{F}})[l^B_i] = ESprices(t^{\mathcal{F}})[l^{A2}_i] - 1$. Therefore, the number of different states according to $\preceq_{ES}$ is polynomial in n.

□

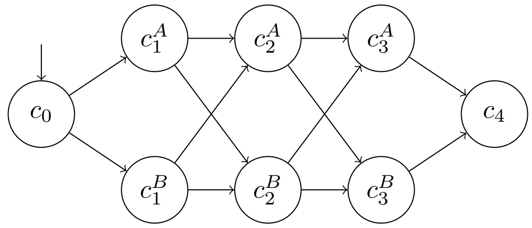
Theorem 9 *$\preceq_S$ is exponentially separated from $\preceq_E$, and therefore also from $\preceq_F$. $\preceq_F$, and therefore also $\preceq_E$, is exponentially separated from $\preceq_S$.*

Proof: First, we prove that $\preceq_F$ can be exponentially better than $\preceq_S$. Consider the following planning task, in which we have a single center variable c (see Figure 3a) with domain $c_1, \dots, c_n$, and c^A_i, c^B_i for $i \in [1, \dots, n-1]$, arranged such that we can move from c_i to c^X_i, and from c^X_i to c_{i+1} ($X \in \{A, B\}$). The initial value is c_1.

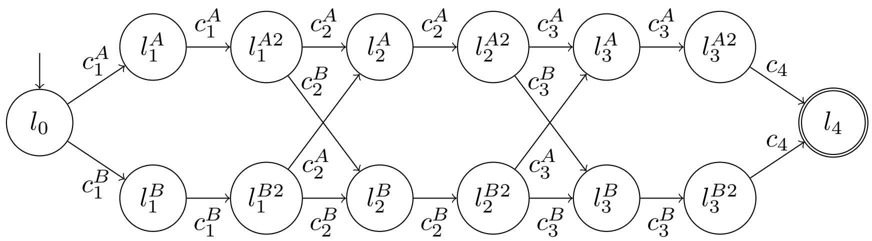
We have a single leaf variable l with values $l_0, \dots, l_n$, and $l^A_i, l^B_i, l^{A2}_i, l^{B2}_i$ for $i \in [1, \dots, n-1]$ with initial value l_0 and goal value l_n. The transitions are depicted in Figure 3b:

- From l_0 to l^X_1 under center precondition c^X_1 for $X \in \{A, B\}$.
- From l^{X2}_{n-1} to l_n under center precondition c^X_n for $X \in \{A, B\}$.
- From each l^X_i to l^{Y2}_i under center precondition c^X_i for $i \in [1, \dots, n-1]$, and $X, Y \in \{A, B\}$.
- From each l^{X2}_i to l^X_{i+1} under center precondition c^X_{i+1} (for $i \in [1, \dots, n-1]$ and $X \in \{A, B\}$).

In this example, the coarsest simulation relation consist only of $s^L \preceq^L s^L_G$ for all s^L. $a_i \not\preceq b_i$ because the center preconditions of their transitions are different. Note that



(a) Center factor



(b) Leaf factor

Figure 2: Illustrative example in which $\preceq_{ES}$ is exponentially better than $\preceq_S$ + $\preceq_E$ for $n = 3$

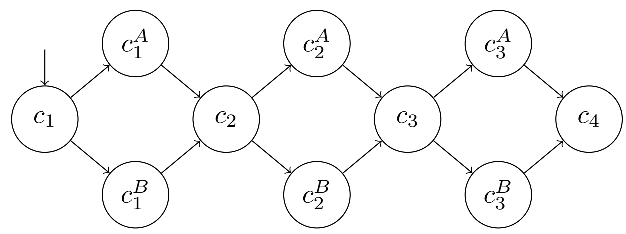
$l_i^A \not\preceq l_i^{A2}$ because $l_i^A \xrightarrow{c_i^A} l_{i+1}^{B2}$ and l_i^{A2} does not have any outgoing transition labeled with c_i^A. l_i^{A2} and l_i^{B2} are always reached at the same time, either via l_i^A or l_i^B. The pricing function remembers whether l_i^A or l_i^B was used and, hence, there are an exponential number of decoupled states with the same center state, even when using $\preceq_S$ pruning.

However, every time l_i^A or l_i^B are reached, the center precondition c_i^A or c_i^B must be satisfied. Both l_i^{A2} and l_i^{B2} have always a price of $prices(s^{\mathcal{F}})[l_i^{A2}] = prices(s^{\mathcal{F}})[l_i^{B2}] \leq prices(s^{\mathcal{F}})[l_i^X] + 1$ for $X \in \{A, B\}$. Therefore, $l_i^A, l_i^B \notin F(prices(s^{\mathcal{F}}))$ for all $s^{\mathcal{F}}$ and i. Since $prices(s^{\mathcal{F}})[l_i^{A2}] = prices(s^{\mathcal{F}})[l_i^{B2}]$, there are only a polynomial number of decoupled states when using dominance pruning with $\preceq_F$.

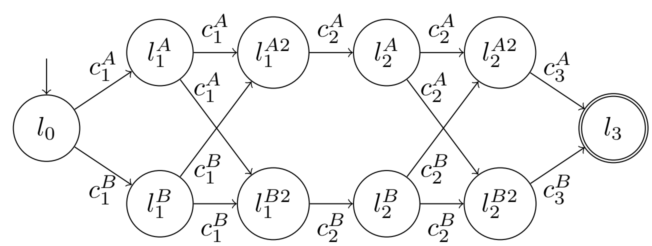
Next, we show that decoupled search with $\preceq_S$ can be exponentially better than decoupled search with $\preceq_E$. Consider a planning task with the same center variable as our previous example. We have a single leaf variable l with values $l_0, \dots, l_n$, with initial value l_0 and goal value l_n. The transitions of the leaf l are depicted in Figure 3c. l has transitions from l_0 to l_1 without any center precondition, from l_0 to l_i under center precondition c_i^A for $i \in [2, n-1]$ and from l_i to l_n under center precondition c_n for $i \in [1, n-1]$.

In this example, decoupled search with $\preceq_B$ expands an exponential number of states with the same center state because the pricing function remembers whether c_i^A was traversed or not for each $i \in [2, \dots, n-1]$. Both $\preceq_F$ and $\preceq_E$ cannot reduce the size of the decoupled state space in this example, because l_n is only reached under center precondition c_n, so $l_i \in F(s^{\mathcal{F}})$ for all $0 < i < n$ and $cs(s^{\mathcal{F}}) \neq c_n$.

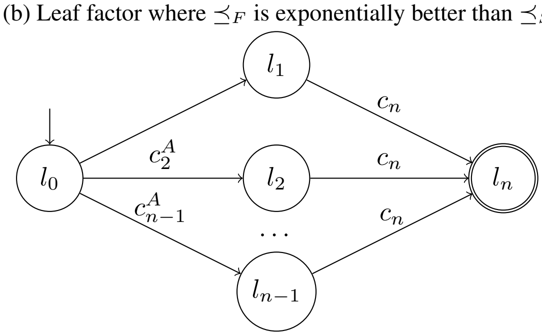
However, since all l_i for $i \in [1, n-1]$ have the same outgoing transitions, $l_i \preceq^L l_1$ for all $i \in [0, n-1]$. As the transition from l_0 to l_1 has no center precondition, $prices(s^{\mathcal{F}})[l_1] = 1$ for any $s^{\mathcal{F}}$. Therefore, $\min_{l_i \preceq^L t^L} prices(s^{\mathcal{F}})[t^L] = 1$ for $i \in [1, n-1]$. Hence, the number of different decoupled states with $\preceq_S$ is polynomial in n. □



(a) Center factor



(b) Leaf factor where $\preceq_F$ is exponentially better than $\preceq_S$.



(c) Leaf factor where $\preceq_S$ is exponentially better than $\preceq_E$.

Figure 3: Exponential separations example for $n = 3$.